(2) assigning a unique one of the plurality of index-addressable temporary storage locations to each one of said plurality of instructions in said instruction buffer, wherein an output corresponding to a given one of said plurality of instructions is stored in said index-addressable temporary storage location assigned to said given one of said plurality of instructions;

(3) determining whether one of said plurality of instructions in said instruction buffer is a dependent instruction, wherein said dependent instruction has an input that is an output of a previous instruction, wherein said previous instruction is an instruction in said instruction buffer that precedes said dependent instruction in the prescribed program order; and

(4) associating said index-addressable temporary storage location assigned to said previous instruction with said input.

B1

2 ~~13~~. The method of claim 1 ~~12~~, further comprising the steps of:

(5) executing said dependent instruction only after said previous instruction produces an output;

(6) storing said output in said index-addressable temporary storage location assigned to said previous instruction; and

(7) performing an operation corresponding to said dependent instruction using said output stored in said index-addressable temporary storage location assigned to said previous instruction as said input.

3 ~~14~~. The method of claim 1 ~~12~~, further comprising the step of associating a done signal with said input, wherein said done signal indicates a status of said previous instruction.

4 ~~15~~. The method of claim 1 ~~12~~, further comprising the step of storing said output in an appropriate register when said previous instruction is retired.

C 5 ~~16~~. A superscalar ~~superscaler~~ processor for executing instructions having a prescribed program order, comprising:

an instruction buffer for storing a plurality of instructions;

34

an index-addressable temporary buffer comprising a plurality of temporary storage locations, wherein each one of said plurality of instructions is assigned to a unique one of said plurality of temporary storage locations, wherein an output corresponding to a given one of said plurality of instructions is stored in said temporary storage location assigned to said given one of said plurality of instructions;

a data dependency checker to locate a dependent instruction stored in said instruction buffer, wherein said dependent instruction has an input that is dependent on a previous instruction, wherein said previous instruction is an instruction in said instruction buffer that precedes said dependent instruction in the prescribed program order; and

a circuit that receives from said data dependency checker dependency data corresponding to said dependent instruction and uses said dependency data to associate said temporary storage location assigned to said previous instruction with said input.

B1

C 1 ~~17.~~ 6 The ~~superscaler~~ superscalar processor of claim ~~16~~ 5, wherein said circuit associates said temporary storage location assigned to said previous instruction with said input by outputting a reference corresponding to said temporary storage location assigned to said previous instruction.

C 1 ~~18.~~ 7 The ~~superscaler~~ superscalar processor of claim ~~17~~ 6, wherein said reference comprises an address.

C 1 ~~19.~~ 8 The ~~superscaler~~ superscalar process of claim ~~18~~ 7, wherein said reference further comprises a 1-bit identifier that indicates whether said address is an address of one of said temporary storage locations or is an address of a register.

C 1 ~~20.~~ 9 The ~~superscaler~~ superscalar processor of claim ~~18~~ 7, wherein said circuit further outputs a first signal indicating a completion status of said previous instruction.

C 1 ~~21.~~ 10 The ~~superscaler~~ superscalar processor of claim ~~20~~ 9, further comprising:

a multiplexer having a plurality of inputs, wherein said multiplexer receives an address of a storage location at each one of said plurality of inputs, wherein one of said addresses received at one of said plurality of inputs is said address included in said reference outputted by said circuit; and

35

an issuer that receives as an input said signal indicating a completion status of said previous instruction and that outputs a select signal to said multiplexer.

11 superscalar 5
22. The ~~superscaler~~ processor of claim ~~16~~, wherein said circuit comprises a priority encoder and a multiplexer, wherein said encoder receives said dependency data, and an output of said encoder is used as a select signal for said multiplexer.

12 superscalar 11
23. The ~~superscaler~~ processor of claim ~~22~~, wherein said multiplexer receives a plurality of references as inputs and outputs a reference corresponding to said output of said encoder, and wherein said reference outputted by said multiplexer represents an address of said temporary storage location assigned to said previous instruction.

13 superscalar 5
24. The ~~superscaler~~ processor of claim ~~16~~, wherein said instruction buffer is capable of storing at most X number of instructions, and said temporary buffer includes at least X number of temporary storage locations, wherein X is a positive integer.

14 superscalar 5
25. The ~~superscaler~~ processor of claim ~~16~~, wherein one of said plurality of instructions is assigned to a unique one of said plurality of storage locations based on a position of said one of said plurality of instructions within said instruction buffer.

15 superscalar 5
26. The ~~superscaler~~ processor of claim ~~16~~, wherein said data dependency checker locates a dependent instruction stored in said instruction buffer by comparing a source register of one of said plurality of instructions to a destination register of each instruction in said instruction buffer that precedes said one of said plurality of instructions in the prescribed program order.

16 superscalar 15
27. The ~~superscaler~~ processor of claim ~~26~~, wherein said data dependency checker comprises a plurality of data dependency circuits, wherein each dependency circuit performs at least on comparison to determine whether any given one of said plurality of instructions depends on a previous instruction.

17 superscalar 16
28. The ~~superscalar~~ processor of claim ~~27~~, wherein one of said data dependency circuits comprises at least one comparator circuit, said comparator circuit receiving a first input and a second input and outputting a dependency signal, said first input corresponding to a source register of a first instruction stored in said instruction buffer and said second input corresponding to a destination register of a second instruction stored in said instruction buffer, wherein said second instruction precedes said first instruction in the prescribed program order, and wherein said dependency signal indicates whether said first instruction is dependent on said second instruction.

18 superscalar 17
29. The ~~superscalar~~ processor of claim ~~28~~, wherein said comparator circuit comprises a plurality of exclusive NOR gates (XNOR) and an AND gate having a plurality of inputs, wherein an output of each XNOR gate is tied to an input of said AND gate, and wherein each XNOR has a first XNOR input and a second XNOR input, wherein each first XNOR input is tied to a bit of said first input and each second XNOR input is tied to a bit of said second input.

19
30. A computer system, comprising:

a memory unit for storing program instructions having a prescribed program order;

a bus for retrieving said program instructions from said memory unit; and

a processor in communication with said bus for executing said program instructions, wherein said processor comprises:

an instruction buffer for storing a plurality of instructions;

an index-addressable temporary buffer comprising a plurality of temporary storage locations, wherein each one of said plurality of instructions is assigned to a unique one of said plurality of temporary storage locations, wherein an output corresponding to a given one of said plurality of instructions is stored in said temporary storage location assigned to said given one of said plurality of instructions;

a data dependency checker to locate a dependent instruction stored in said instruction buffer, wherein said dependent instruction has an input that is dependent on a previous instruction, wherein said previous instruction is an instruction in said instruction buffer that precedes said dependent instruction in the prescribed program order; and

a circuit that receives from said data dependency checker dependency data corresponding to said dependent instruction and uses said dependency data to associate said temporary storage location assigned to said previous instruction with said input.

20 ~~31~~. The computer system of claim 19 ~~30~~, wherein said circuit associates said temporary storage location assigned to said previous instruction with said input by outputting an address of said temporary storage location assigned to said previous instruction.

21 ~~32~~. The computer system of claim 20 ~~31~~, wherein said circuit further outputs a first signal indicating a completion status of said previous instruction.

22 ~~33~~. The computer system of claim 21 ~~32~~, further comprising:

a multiplexer having a first plurality of inputs, wherein said multiplexer receives an address of a storage location at each one of said plurality of inputs, wherein one of said addresses received at one of said plurality of inputs is said address outputted by said circuit; and

an issuer that receives as an input said signal indicating a completion status of said previous instruction, and that outputs a select signal to said multiplexer.

23 ~~34~~. The computer system of claim 19 ~~30~~, wherein said circuit comprises a priority encoder and a multiplexer, wherein said encoder receives said dependency data, and an output of said encoder is used as a select signal for said multiplexer.

24 ~~35~~. The computer system of claim 23 ~~34~~, wherein said output of said encoder represents said previous instruction.

25 ~~36~~. The computer system of claim 24 ~~35~~, wherein said multiplexer receives a plurality of references as inputs and outputs a reference corresponding to said output of said encoder, wherein said reference outputted by said multiplexer represents an address of said temporary storage location assigned to said previous instruction.

24 19
~~37~~. The computer system of claim ~~30~~, wherein said instruction buffer is capable of storing at most X number of instructions, and said temporary buffer includes at least X number of temporary storage locations, wherein X is a positive integer.

27 19
~~38~~. The computer system of claim ~~30~~, wherein one of said plurality of said program instructions is assigned to a unique one of said plurality of storage locations based on a position of said one of said plurality of said program instructions within said instruction buffer.

28 19
~~39~~. The computer system of claim ~~30~~, wherein said data dependency checker locates a dependent instruction stored in said instruction buffer by comparing a source register of one of said plurality of said program instructions to a destination register of each instruction in said instruction buffer that precedes said one of said plurality of said program instructions in the prescribed program order.

29 28
~~40~~. The computer system of claim ~~39~~, wherein said data dependency checker comprises a plurality of data dependency circuits, wherein each dependency circuit performs at least on comparison to determine whether any given one of said plurality of said program instructions depends on a previous instruction.

30 29
~~41~~. The computer system of claim ~~40~~, wherein one of said data dependency circuits comprises at least one comparator circuit, said comparator circuit receiving a first input and a second input and outputting a dependency signal, said first input corresponding to a source register of a first instruction stored in said instruction buffer and said second input corresponding to a destination register of a second instruction stored in said instruction buffer, wherein said second instruction precedes said first instruction in the prescribed program order, and wherein said dependency signal indicates whether said first instruction is dependent on said second instruction.

31 30
~~42~~. The computer system of claim ~~41~~, wherein said comparator circuit comprises a plurality of exclusive NOR gates (XNOR) and an AND gate having a plurality of inputs, wherein an output of each XNOR gate is tied to an input of said AND gate, and wherein each XNOR has a first

XNOR input and a second XNOR input, wherein each first XNOR input is tied to a bit of said first input and each second XNOR input is tied to a bit of said second input.

32. A ~~superscaler~~ superscalar processor for executing instructions having a prescribed program order, comprising:

an instruction buffer storing a plurality of instructions;

a register file having a plurality of registers and a plurality of index-addressable temporary storage locations, wherein each one of said plurality of instructions is assigned to a unique one of said plurality of temporary storage locations and one of said plurality of registers, wherein an output corresponding to a given one of said plurality of instructions is stored in said temporary storage location assigned to said given one of said plurality of instructions;

a data dependency checker to locate dependent instructions stored in said instruction buffer, wherein a dependent instruction is an instruction that should not be executed until after a particular previous instruction within said instruction buffer is executed, wherein said data dependency checker outputs dependency information corresponding to each instruction stored in said instruction buffer;

a plurality of circuits, wherein each of said plurality of circuits corresponds to an instruction in said instruction buffer and receives dependency data from said data dependency checker, wherein said dependency data received at a particular circuit corresponds to said instruction to which said particular circuit corresponds, and wherein a circuit corresponding to a given dependent instruction outputs a reference representing said temporary storage location assigned to a particular previous instruction that must be executed prior to said given dependent instruction.

33. A computer system, comprising:

a memory unit for storing program instructions having a prescribed program order;

a bus for retrieving said program instructions from said memory unit; and

a processor in communication with said bus for executing said program instructions, wherein said processor comprises:

an instruction buffer storing a plurality of instructions;

a register file having a plurality of registers and a plurality of index-addressable temporary storage locations, wherein each one of said plurality of instructions is assigned to a unique one of said plurality of temporary storage locations and one of said plurality of registers, wherein an output corresponding to a given one of said plurality of instructions is stored in said temporary storage location assigned to said given one of said plurality of instructions;

a data dependency checker to locate dependent instructions stored in said instruction buffer, wherein a dependent instruction is an instruction that should not be executed until after a particular previous instruction within said instruction buffer is executed, wherein said data dependency checker outputs dependency information corresponding to each instruction stored in said instruction buffer; and

a plurality of circuits, wherein each of said plurality of circuits corresponds to an instruction in said instruction buffer and receives dependency data from said data dependency checker, wherein said dependency data received at a particular circuit corresponds to said instruction to which said particular circuit corresponds, and wherein a circuit corresponding to a given dependent instruction outputs a reference representing said temporary storage location assigned to a particular previous instruction that must be executed prior to said given dependent instruction.

34
~~45~~. In a computer system having a register file comprising a plurality of registers and a plurality of index-addressable temporary storage locations, a method for executing instructions having an input and an output and having a prescribed program order, comprising the steps of:

(1) assigning a unique one of the plurality of index-addressable temporary storage locations to each one of a plurality of instructions in an instruction buffer, wherein the output corresponding to a given one of said plurality of instructions is stored in said temporary storage location assigned to said given one of said plurality of instructions;

(2) determining whether one of said plurality of instructions in said instruction buffer is a dependent instruction, wherein said dependent instruction has an input that is dependent on a previous instruction, wherein said previous instruction is an instruction in said instruction buffer that precedes said dependent instruction in the prescribed program order; and

41